

THE INFORMATION FLOW FRAMEWORK: NEW ARCHITECTURE

ROBERT E. KENT

“Philosophy cannot become scientifically healthy without an immense technical vocabulary. We can hardly imagine our great-grandsons turning over the leaves of this dictionary without amusement over the paucity of words with which their grandsires attempted to handle metaphysics and logic. Long before that day, it will have become indispensably requisite, too, that each of these terms should be confined to a single meaning which, however broad, must be free from all vagueness. This will involve a revolution in terminology; for in its present condition a philosophical thought of any precision can seldom be expressed without lengthy explanations.” — Charles Sanders Peirce, Collected Papers 8:169

ABSTRACT. The Information Flow Framework (IFF) uses institution theory as a foundation for the semantic integration of ontologies. It represents metalogic, and as such operates at the structural level of ontologies. The content, form and experience of the IFF could contribute to the development of a standard ontology for category theory. The foundational aspect of the IFF helps to explain the relationship between the fundamental concepts of set theory and category theory. The development of the IFF follows two design principles: conceptual warrant and categorical design. Both are limitations of the logical expression. Conceptual warrant limits the content of logical expression, by requiring us to justify the introduction of new terminology (and attendant axiomatizations). Categorical design limits the form of logical expression (of all mathematical concepts and constraints) to atomic expressions: declarations, equations or relational expressions. The IFF is a descriptive category metatheory. It is descriptive, since it follows the principle of conceptual warrant; it is categorical, since it follows the principle of categorical design; and it is a metatheory, since it provides a framework for all theories.

2000 Mathematics Subject Classification: 18A15.

Key words and phrases: category, metatheory, ontology, metalanguage, institution, standard.

© Robert E. Kent, 2007. Permission to copy for private use granted.

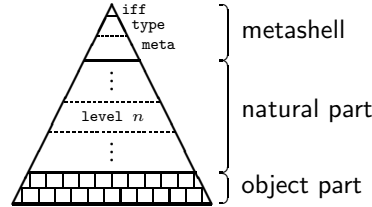


Figure 1: IFF Architecture (iconic)

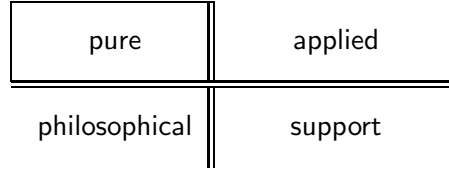


Figure 2: Roles in Category Theory

1. Introduction

The Information Flow Framework (IFF) is a descriptive category metatheory [7] currently under development that provides an important practical application of category theory [10] to knowledge representation, knowledge maintenance and the semantic web ([5],[6]). The categorical approach of the IFF provides a principled framework for the modular design and semantic integration of object-level ontologies. The IFF forms the structural aspect of the IEEE P1600.1 Standard Upper Ontology (SUO) project¹. It is an experiment in foundations, which follows a bottom-up approach to logical description. A preliminary description of the IFF was presented at the International Category Theory Conference in Vancouver in 2004 [7]. This paper discusses a new, modular, more mature architecture (the transition from the preliminary description to this more mature architecture is discussed in subsubsection 2.2.3).

1.1. BACKGROUND. Before discussing the architecture, we discuss: roles in category theory, ontologies with examples, development, design principles and current state.

1.1.1. ROLES. All category theory activities involve several roles (Figure 2): pure, applied, philosophical and support. The pure role is to develop category theory. This is the central role. An example of this role was played by the International Category Theory Conference at White Point in 2006. The applied role uses category theory in various applications such as mathematics, programming languages, concurrency and knowledge engineering. The philosophical role tries to explain and/or justify category theory. This may be based on various historical and social forces, or its position in reality. At any particular time, an activity may emphasize a certain role. The support role is for the implementation of category theory. This will aid the working category theorist. Support might involve the development of suitable ontologies, logical code, grammars and programming code. The IFF development has involved several of these roles. Initially, its goal was the application of category theory to knowledge engineering. More recently, the IFF has reverse this approach, and now seeks to support category theory by applying some of the tools and techniques of knowledge engineering. Since this recent supporting role involves foundations, the IFF is implicitly involved to a certain extent in a philosophical role.

¹The main IFF webpage is located at <http://suo.ieee.org/IFF/>.

1.1.2. ONTOLOGIES. The term ‘ontology’ was first coined in the 17th century. It comes from two Greek words, *οντος* meaning *of being* and *λογία* meaning *science, study* or *theory*. The ancient Greek philosopher Aristotle defined ontology to be “the science of being qua (in the capacity of) being”; hence, ontology is the science of being inasmuch as it is being, or the study of beings insofar as they exist. The dictionary (Merriam-Webster) defines ontology to be 1 : a branch of metaphysics concerned with the nature and relations of being 2 : a particular theory about the nature of being or the kinds of existents. The Encyclopedia (Brittanica) describes ontology to be the theory or study of being as such; i.e., of the basic characteristics of all reality. Ontology is synonymous with metaphysics or “first philosophy” as defined by Aristotle. Artificial Intelligence and Knowledge Engineering define an ontology² to be a formal, explicit specification of a shared conceptualization. It is *an abstract model of some phenomena in the world* (semantic conceptualization), explicitly represented as *concepts, relationships and constraints* (logic-oriented), which is *machine-readable* (formal and explicit) and incorporates *the consensual knowledge of some community* (shared and relative).

1.1.3. EXAMPLES. Figure 3 illustrates an ontology of roadmaps. In the area of biology, the Gene Ontology (GO)³ is an actual example of a functioning ontology, representing concepts such as gene, protein and metabolic pathway, and predicates such as being a regulator gene. A possible ontology for the category theory community might represent concepts such as category or adjunction, predicates such as small-complete, functions such as the object/morphism set function of a category, and relations such as the subcategory order or the composable relation for functors.

1.2. THE IFF. The IFF originated from a desire to use category theory for the representation and semantic integration of ontologies.

1.2.1. DEVELOPMENT. The IFF is being develop under the auspices of the IEEE Standard Upper Ontology (SUO) project [11]. It was the first of several approved SUO resolutions. There was always a close connection [8] between the goals of the IFF and the theory of institutions [3]. There was also a connection to foundations, since from the category-theoretic perspective, a strong requirement of the IFF formalism was the complete incorporation of various structures in a large category \mathcal{C} , such as the pullback square that defines the source of the composition map

$$\begin{array}{ccc} \text{mor}(\mathcal{C}) \times_{\text{obj}(\mathcal{C})} \text{mor}(\mathcal{C}) & \xrightarrow{\pi_1^{\mathcal{C}}} & \text{mor}(\mathcal{C}) \\ \pi_0^{\mathcal{C}} \downarrow & \lrcorner & \downarrow \partial_0^{\mathcal{C}} \\ \text{mor}(\mathcal{C}) & \xrightarrow{\partial_1^{\mathcal{C}}} & \text{obj}(\mathcal{C}) \end{array}$$

1.2.2. DESIGN PRINCIPLES. During the IFF development, two design principles have emerged as important: conceptual warrant and categorical design.

²Ontologies can be thought of as taxonomies, logical theories or knowledge-bases.

³located at <http://www.geneontology.org/>

Concepts = Types = Entities

- highway = road
- geographical-feature
 - location = point
 - * exit
 - * interchange
 - * town
 - * rest-area
 - line = linear-feature
 - * creek
 - * river
 - * railroad
 - area
 - * lake
 - * mountain
 - * city
 - * county
 - * state = province
 - * country
- territorial-division
 - county
 - state
 - country
- urban-area
 - town
 - city

Predicates = Parts

- principal : highway
- toll-road : highway
- freeway : highway
- scenic : highway
- is-capital : urban-area

Functions = Maps

- name(number) : highway \rightarrow name-tag \times number
 - number-of-lanes : highway \rightarrow number
 - distance : point \times point \rightarrow number
 - facility : rest-area \rightarrow facility-tag
 - intersection : ext(crosses) \rightarrow point
 - exit-location : exit \rightarrow highway \times number
 - lies-in : county \rightarrow state
-
- name-tag = {interstate, state, county}
 - facility-tag = {full, partial, none}

Relations

- crosses : line \rightarrow line
- traverses : highway \rightarrow territorial-division
- goes-through : road \rightarrow urban-area

Axioms

- $\forall (x, y \in \text{linear-feature}) (\text{crosses}(x, y) \Rightarrow \text{crosses}(y, x))$
- $\forall (h \in \text{highway}, c \in \text{county}, s \in \text{state}) ((\text{traverses}(h, c) \ \& \ \text{lies-in}(c, s)) \Rightarrow \text{traverses}(h, s))$
- $\forall (x, y, z \in \text{location}) (\text{distance}(x, z) \leq \text{distance}(x, y) + \text{distance}(y, z))$

Figure 3: An Ontology of Roadmaps

- Conceptual Warrant: [content]⁴ *IFF terminology requires conceptual warrant.* Warrant means evidence for or token of authorization. Conceptual warrant is an adaptation of the librarianship notion of literary warrant. According to the Library of Congress (LOC), its collections serve as the literary warrant (i.e., the literature on which the controlled vocabulary is based) for the LOC subject headings system. Likewise for the IFF, its lower or more peripheral axiomatized concepts serve as the conceptual warrant for the higher level IFF terminology; hence, any term should reference a concept needed in a lower metalevel or more peripheral axiomatization.

LOC	subject headings	collections
IFF	higher terms	lower concepts

- Categorical Design: [form] *IFF module design should follow good category-theoretic intuitions.* Axiomatizations should complete any implicit ideas. For example, any implicit adjunctions should be formalized explicitly. Any current axiomatization may only be partially completed. Axiomatizations should be atomic. Thus, axiomatizations should be in the form of declarations, equations and relational expressions. No axioms should use explicit logical notation: no variables, quantifications or logical connectives should be used. Although the metashell axiomatization uses first order expression, the natural part axiomatization is atomic.

1.2.3. CURRENT STATE. The IFF development is constantly under revision. Attention and activity has moved from applications of institution theory to a category theory standard. Several concepts about development have emerged: the two design principles; an architectural framework; and three concurrent development processes (Figure 4 indicates degree of completion): axiomatic expression (natural language \Rightarrow first order \Rightarrow atomic); category (finitely complete \Rightarrow cartesian-closed \Rightarrow topos); and element (ordinary \Rightarrow generalized).

2. Architecture

*“Such is ‘set theory’ in the practice of mathematics;
it is part of the essence from which organization emerges.”*
~ F.W. Lawvere

The new IFF architecture (Figure 5) — a two dimensional structure consisting of levels (the vertical dimension), namespaces (the horizontal dimension) and meta-ontologies (coherent composites of namespaces) — is described in terms of parts, aspects and components. The vertical dimension of the IFF Architecture consists of three parts: the objective part ($n = 0$), the natural part ($1 \leq n < \infty$) and the supranatural part ($n \in \{\text{meta, type, iff}\}$).

⁴A non-starter during the IFF development was a topos axiomatization. This received objections from the SUO working group, in part due to its lack of support by motivating examples. Rejection of the topos axiomatization prompted the idea of conceptual warrant.

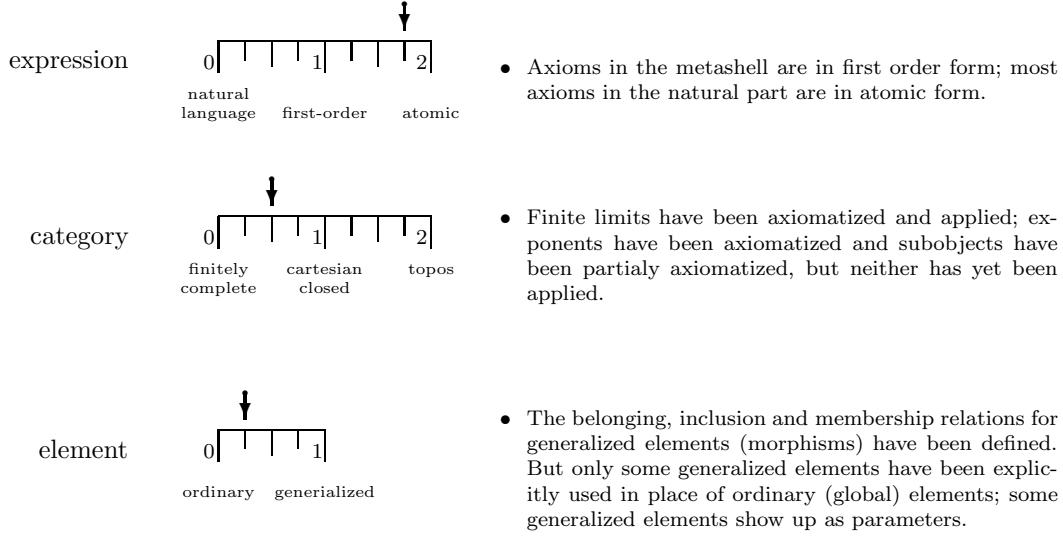


Figure 4: Current Development State

Along the horizontal dimension, the natural part is partitioned into pure and applied aspects, and the pure aspect is partitioned into core and structural components. The pure aspect, which contains meta-ontologies axiomatizing in adjunctive form the set-theoretic and category-theoretic foundations needed elsewhere in the IFF, is the first step towards a standard ontology for category theory. The kernel of the core component, called the IFF metastack, represents a chain of toposes⁵ $\text{Set} = \langle \text{Set}_1 \subset \text{Set}_2 \subset \dots \subset \text{Set}_n \subset \dots \rangle$ anchoring the entire IFF architecture. The metastack structure is based upon Cantor's diagonal argument. The applied aspect contains meta-ontologies providing the terminology and axiomatization needed for the logical and semiotic functionality in applications.

2.1. MODULAR STRUCTURE. The modular structure of the IFF architecture consists of parts, aspects and components.

2.1.1. METASHELL. The supranatural part is also called the IFF metashell. The form of the metashell axiomatization has a first order expression with explicit logical structure. The content of the metashell axiomatization consists of (only) one namespace at each of the three metalevels {meta,type,iff}: the iff level contains the **IFF-IFF** namespace, the **type** level contains the **IFF-TYPE** namespace, and the **meta** level contains the **IFF-META** namespace. The **IFF-IFF** namespace axiomatizes a directed graph of abstract sets and functions⁶, the **IFF-TYPE** namespace axiomatizes a finitely-complete category of abstract

⁵A category \mathcal{E} is a topos when it has finite limits, it is cartesian closed, and it has a subobject classifier; equivalently, when it has finite limits and comes equipped with an object of truth values $\Omega_{\mathcal{E}} \in \text{obj}(\mathcal{E})$, a power function $\wp_{\mathcal{E}} : \text{obj}(\mathcal{E}) \rightarrow \text{obj}(\mathcal{E})$ where $\text{Sub}_{\mathcal{E}} A$ is the set of subobjects of A , and two natural isomorphisms $\text{Sub}_{\mathcal{E}}(A) \cong \text{Hom}_{\mathcal{E}}[A, \Omega_{\mathcal{E}}]$ and $\text{Hom}_{\mathcal{E}}[A \times B, \Omega_{\mathcal{E}}] \cong \text{Hom}_{\mathcal{E}}[A, \wp_{\mathcal{E}} B]$.

⁶It contains only the five terms {'thing', 'set', 'function', 'source', 'target'}.

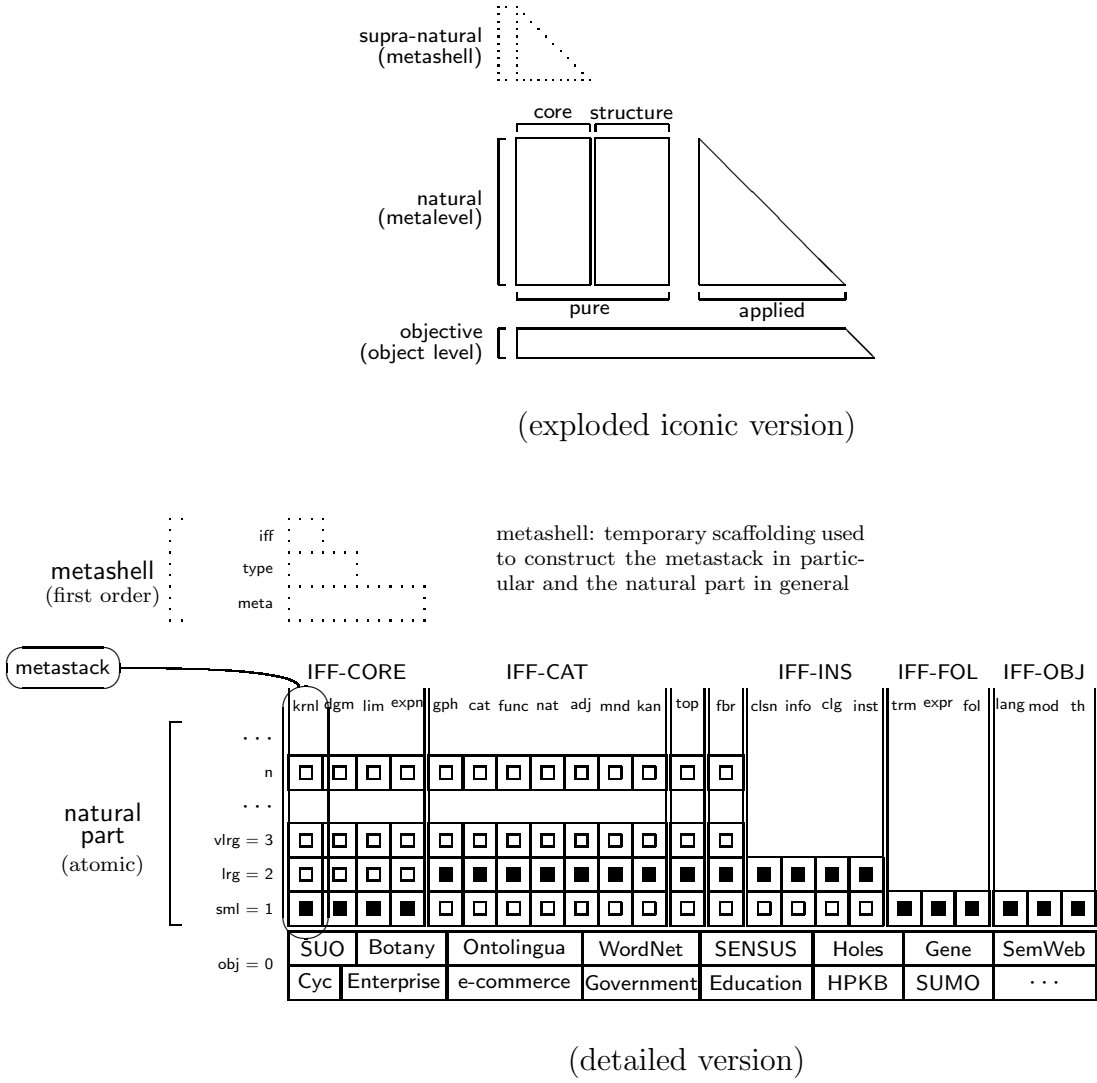


Figure 5: The IFF Architecture

	...	
$\text{Set}_n \in \text{obj}(\text{Cat}_{n+1})$	$\text{Cat}_{n+1} = \text{cat}(\text{Set}_{n+1})$	level $n+1$ categories
$\text{Set}_{n-1} \in \text{obj}(\text{Cat}_n)$	$\text{Cat}_n = \text{cat}(\text{Set}_n)$	level n categories
	...	
$\text{Set}_1 \in \text{obj}(\text{Cat}_2)$	$\text{Cat}_2 = \text{cat}(\text{Set}_2)$	large categories
	$\text{Cat}_1 = \text{cat}(\text{Set}_1)$	small categories

Table 1: Topos Chain

sets and functions⁷, and the **IFF-META** namespace axiomatizes a topos of Cantorian featureless abstract sets and functions. From one point of view, the metashell serves as a temporary scaffolding for construction of the entire IFF architecture in the natural part.

2.1.2. OBJECTIVE PART. The form of the objective part axiomatization has a simple atomic expression with no logical structure. The content of the objective part axiomatization consists of terminology for object-level ontologies. The form of the natural part axiomatization also has a simple atomic expression with some implicit logical structure. The content of the natural part axiomatization consists of namespaces representing many of the basic concepts of mathematics and logic. By conforming to the principle of categorical design, the logical expression in the natural part is atomic, thus using commutative diagrams to express the intuitions and ideas of the working category theorist.

2.1.3. PURE ASPECT. The pure aspect of the IFF architecture is partitioned into a core component and a structural component. The core component represents set theory. The structural component represents category theory. The set-theoretic and category-theoretic axiomatizations constrain each other: any category is defined as a set of objects and a set of morphisms with connecting functions; whereas the collections of sets and functions (at any metalevel) form a category. The axiomatization for any concept in the pure aspect is given in one generic module (namespace) at level n , for $1 \leq n < \infty$. The finite metalevels, $1 \leq n < \infty$, are populated by generic and parametric meta-ontologies. Generic means that the terminology and axiomatization for any two metalevels is identical. Parametric means that the metalevel index is a parameter. Only one copy of a meta-ontology with a level parameter is needed for all finite levels.

2.1.4. CORE COMPONENT. The core component of the IFF architecture contains a single generic level n meta-ontology **IFF-SET** for set theory, which incorporates the specialization of either the level $n+1$ version of itself and/or the **IFF-META** namespace from the metashell. The **IFF-SET** meta-ontology specifies set theory as a chain of toposes (Table 1) of Cantorian featureless abstract sets

$$\text{Set} = \langle \text{Set}_1 \subset \text{Set}_2 \subset \cdots \subset \text{Set}_n \subset \cdots \rangle$$

where Set_1 contains “small” sets and functions between “small” sets and Set_2 contains “large” sets and functions between “large” sets. This chain is motivated by and compatible with the Cantorian expansion of sets discussed in subsection 3.2.

⁷It contains approximately 450 terms, with 250 terms in the kernel subnamespace and 100 terms each in the diagram and limit subnamespaces.

IFF Term	Concept
#n:set	Set_n
#n.set:set	$\text{obj}(\text{Set}_n)$
#n.set:power	$\text{obj}(\mathcal{P}_{\text{Set}_n})$
#n.set:{zero, one, two}	$0_{\text{Set}_n}, 1_{\text{Set}_n}, 2_{\text{Set}_n} = \Omega_{\text{Set}_n}$
#n.ftn:function	$\text{mor}(\text{Set}_n)$
#n.ftn:power	$\text{mor}(\mathcal{P}_{\text{Set}_n})$
#n.ftn:composition	\circ_{Set_n}
#n.ftn:identity	1_{Set_n}
#n.pred:fiber	$\text{Sub}_{\text{Set}_n}$
#n.pred:binary-meet	\wedge_{Set_n}
#n.rel:fiber01	$\varphi_{01}^{\text{Set}_n}$
#n.lim.prd2.obj:product	\times_{Set_n}
#n.lim.prd2.obj:projection{0,1}	$\pi_i^{\text{Set}_n} \quad i = 0, 1$
#n.exp.obj:exponent	B^A
#n.exp.obj:evaluation	$B^A \times A \rightarrow B$
#n.exp.obj:hom	$\text{Set}_n[-, -]$
#n.exp.obj:curry	$\text{Set}_n[C \times A, B] \rightarrow \text{Set}_n[C, B^A]$

Table 2: Set_n as a Topos

2.1.5. STRUCTURE COMPONENT. The structure component of the IFF architecture contains various generic meta-ontologies for category theory, (IFF-CAT, IFF-2CAT, IFF-DCAT, ...). By axiomatizing the basic concepts of category theory, such as categories, functors, natural transformations, adjunctions and monads, the IFF-CAT meta-ontology specifies category theory as a chain (Table 1) of internal categories

$$\text{Cat} = \langle \text{Cat}_1 \subset \text{Cat}_2 \subset \cdots \subset \text{Cat}_n \subset \cdots \rangle$$

in the toposes Set . The generic level axiomatizations for the selected topos-representing terms in Table 2 justify the assertion: for each n , the category Set_n of level n sets and functions is a topos.

2.1.6. ATOMIC EXPRESSION. To reiterate, the natural part of the IFF is where most of the concepts and constraints of mathematics and logic are represented. This representation is atomic, consisting of either declarations or equations. The declarations state that an element is in a set or a pair of elements satisfies a relation. The equations, which have only unary functions in their terms, are equivalent to the commutative diagrams of category theory. Hence, in the IFF all of the axiomatizations of category theory, set theory, logic, institution theory, science, et cetera, are in the form of declarations or commutative diagrams. This is in contrast to the usual notion that category-theoretic representations are first order.

2.2. OTHER FEATURES. Other features of the IFF architecture include: the meta-stack, a simplification that erases links to the metashell, the transition from a previous architecture [7], the IFF grammar, and the IFF as hierarchical metalanguage.

$$\begin{array}{ccccccc}
\cdots & \sqsubseteq & \mathbf{set}_n & \sqsubseteq & \mathbf{set}_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \mathbf{set}_{\mathbf{meta}} & \sqsubseteq & \mathbf{set}_{\mathbf{type}} & \sqsubseteq & \mathbf{set}_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \mathbf{ftn}_n & \sqsubseteq & \mathbf{ftn}_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \mathbf{ftn}_{\mathbf{meta}} & \sqsubseteq & \mathbf{ftn}_{\mathbf{type}} & \sqsubseteq & \mathbf{ftn}_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \mathbf{pred}_n & \sqsubseteq & \mathbf{pred}_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \mathbf{pred}_{\mathbf{meta}} & \sqsubseteq & \mathbf{pred}_{\mathbf{type}} & \sqsubseteq & \mathbf{pred}_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \mathbf{rel}_n & \sqsubseteq & \mathbf{rel}_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \mathbf{rel}_{\mathbf{meta}} & \sqsubseteq & \mathbf{rel}_{\mathbf{type}} & \sqsubseteq & \mathbf{rel}_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \mathbf{endo}_n & \sqsubseteq & \mathbf{endo}_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \mathbf{endo}_{\mathbf{meta}} & \sqsubseteq & \mathbf{endo}_{\mathbf{type}} & \sqsubseteq & \mathbf{endo}_{\mathbf{iff}}
\end{array}$$

Subset

$$\begin{array}{ccccccc}
\cdots & \sqsubseteq & \partial_0^n & \sqsubseteq & \partial_0^{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \partial_0^{\mathbf{meta}} & \sqsubseteq & \partial_0^{\mathbf{type}} & \sqsubseteq & \partial_0^{\mathbf{iff}} \\
\cdots & \sqsubseteq & \partial_1^n & \sqsubseteq & \partial_1^{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \partial_1^{\mathbf{meta}} & \sqsubseteq & \partial_1^{\mathbf{type}} & \sqsubseteq & \partial_1^{\mathbf{iff}} \\
\cdots & \sqsubseteq & \delta_n & \sqsubseteq & \delta_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \delta_{\mathbf{meta}} & \sqsubseteq & \delta_{\mathbf{type}} & \sqsubseteq & \delta_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \varepsilon_n & \sqsubseteq & \varepsilon_{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \varepsilon_{\mathbf{meta}} & \sqsubseteq & \varepsilon_{\mathbf{type}} & \sqsubseteq & \varepsilon_{\mathbf{iff}} \\
\cdots & \sqsubseteq & \sigma_0^n & \sqsubseteq & \sigma_0^{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \sigma_0^{\mathbf{meta}} & \sqsubseteq & \sigma_0^{\mathbf{type}} & \sqsubseteq & \sigma_0^{\mathbf{iff}} \\
\cdots & \sqsubseteq & \sigma_1^n & \sqsubseteq & \sigma_1^{n+1} & \sqsubseteq & \cdots & \sqsubseteq & \sigma_1^{\mathbf{meta}} & \sqsubseteq & \sigma_1^{\mathbf{type}} & \sqsubseteq & \sigma_1^{\mathbf{iff}}
\end{array}$$

$$\begin{array}{ccccccc}
\cdots & \dot{\sqsubseteq} & (\partial_{i=0,1}^n) & \dot{\sqsubseteq} & (\partial_{i=0,1}^{n+1}) & \dot{\sqsubseteq} & \cdots & \dot{\sqsubseteq} & (\partial_{i=0,1}^{\mathbf{meta}}) & \dot{\sqsubseteq} & (\partial_{i=0,1}^{\mathbf{type}}) & \dot{\sqsubseteq} & (\partial_{i=0,1}^{\mathbf{iff}}) \\
\cdots & \dot{\sqsubseteq} & \gamma_n & \dot{\sqsubseteq} & \gamma_{n+1} & \dot{\sqsubseteq} & \cdots & \dot{\sqsubseteq} & \gamma_{\mathbf{meta}} & \dot{\sqsubseteq} & \gamma_{\mathbf{type}} & \dot{\sqsubseteq} & \gamma_{\mathbf{iff}} \\
\cdots & \dot{\sqsubseteq} & (\sigma_{i=0,1}^n) & \dot{\sqsubseteq} & (\sigma_{i=0,1}^{n+1}) & \dot{\sqsubseteq} & \cdots & \dot{\sqsubseteq} & (\sigma_{i=0,1}^{\mathbf{meta}}) & \dot{\sqsubseteq} & (\sigma_{i=0,1}^{\mathbf{type}}) & \dot{\sqsubseteq} & (\sigma_{i=0,1}^{\mathbf{iff}})
\end{array}$$

(Optimal-)Restriction

$$\begin{array}{ccccccc}
\cdots & \preceq & \sqsubseteq_n & \preceq & \sqsubseteq_{n+1} & \preceq & \cdots & \preceq & \sqsubseteq_{\mathbf{meta}} & \preceq & \sqsubseteq_{\mathbf{type}} & \preceq & \sqsubseteq_{\mathbf{iff}} \\
\cdots & \preceq & \leq_n & \preceq & \leq_{n+1} & \preceq & \cdots & \preceq & \leq_{\mathbf{meta}} & \preceq & \leq_{\mathbf{type}} & \preceq & \leq_{\mathbf{iff}} \\
\cdots & \preceq & \sqsubseteq_n & \preceq & \sqsubseteq_{n+1} & \preceq & \cdots & \preceq & \sqsubseteq_{\mathbf{meta}} & \preceq & \sqsubseteq_{\mathbf{type}} & \preceq & \sqsubseteq_{\mathbf{iff}} \\
\cdots & \preceq & \preceq_n & \preceq & \preceq_{n+1} & \preceq & \cdots & \preceq & \preceq_{\mathbf{meta}} & \preceq & \preceq_{\mathbf{type}} & \preceq & \preceq_{\mathbf{iff}}
\end{array}$$

Abridgment

Table 3: Kernel Chains

2.2.1. METASTACK. The IFF metastack, which is the kernel of the core component, represents the Cantorian expansion. It binds and anchors the natural part of the IFF, and connects the natural part to the metashell. The metastack contains a lattice-like structure for the fundamental subset (\sqsubseteq), restriction ($\dot{\sqsubseteq}$) and optimal-restriction ($\dot{\sqsubseteq}$), delimitation (\leq) and abridgment (\preceq) orders on sets, functions, predicates and relations, respectively. This is illustrated in Figure 3, where ∂_i are the source and target for functions with pairing (∂_i), γ and δ are the genus and differentia for predicates, σ_i are the component sets for relations with pairing (σ_i), and ε is the extent function for relations. Just as (binary) relations are predicates (unary relations or parts) on a binary product and predicates are special functions (injections), so also abridgment is a special case of delimitation and delimitation is a special case of optimal-restriction.

2.2.2. SIMPLIFICATION. Consider, amongst others, the level n source function $\partial_0^n : \mathbf{ftn}_n \rightarrow \mathbf{set}_n$, a level $n+1$ function. We intend this to be a restriction of both the level $n+1$ source function $\partial_0^n \sqsubseteq \partial_0^{n+1}$ and the **meta** level source function $\partial_0^n \sqsubseteq \partial_0^{\mathbf{meta}}$. But since restriction defines the smaller function, the latter link is clearly redundant. The links, such as the latter, for the basic metastack orders (subset, function (optimal-)restriction, predicate delimitation and relation abridgment) between the natural part and the metashell are special. These were used in the experimental phase of IFF development to help develop the axiomatization for the natural part into a first order expression. But

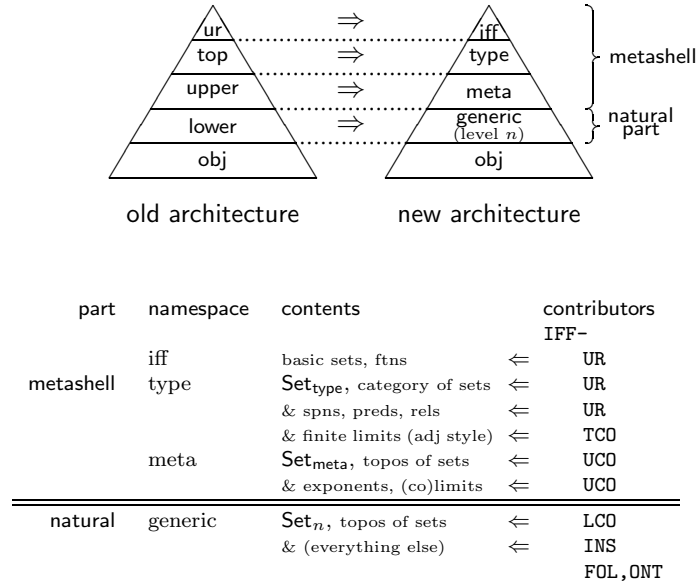


Figure 6: Transition

as the principle of categorical design pushed towards an atomic expression for the natural part, the links to the metashell were needed less and less. With full atomic expression for the natural part, it appears that we can make a choice whether to use these or not. A minimal axiomatization would not need these. It appears that with the full atomic expression they are only needed for added justification. A set-theorist, such as Feferman [2], who believes in the preeminence of the basic set-theoretic notions of “collection” and “operation” (represented in the IFF by the set and function collections in the metashell, in particular those at the *iff* metalevel), may want these links to remain in place. But the IFF is intended to have a category-theoretic foundation. Thus, we can choose whether or not to keep these links. This is why we say the metashell is a temporary scaffolding used to construct the natural part.

2.2.3. TRANSITION. The new architecture evolved through much trial-and-error experimentation. Much of the axiomatization for the metashell has been taken from some core IFF meta-ontologies in the old architecture. In fact, loosely speaking, we have effectively raised the previous four-tiered (ur-top-upper-lower) architecture of the old core to become the new metashell (Figure 6). But, this happened somewhat accidentally with no a prior design. We have been guided only by the requirements. And apparently these requirements have intrinsically remained the same, but only shifted upwards in the metalevel hierarchy. For the IFF-*IFF* namespace, we have used the recent version of the IFF Ur (meta) Ontology (July 2004)⁸ (but simplified, since we have moved introduction of the predicate and relation concepts down to the type level). For the type kernel

⁸ located at <http://suo.ieee.org/IFF/metastack/UR.pdf>

		Math	IFF
names			
		a	o.i:a inner context i outer context o
atoms and terms			
set element	$x \in X$ or $X(x)$	$(X \ x)$	
predicate member	$x \in b$ or $b(x)$	$(b \ x)$	
relation member	$(x, y) \in r$ or $r(x, y)$	$(r \ x \ y)$	
function application	$f(x)$	$(f \ x)$	
equations			
	$\sigma = \tau$ or $=(\sigma, \tau)$	$(= \ s \ t)$	
connectives			
conjunction	$\phi \wedge \psi$ or $\wedge(\phi, \psi)$	$(\text{and } P \ Q)$	
disjunction	$\phi \vee \psi$ or $\vee(\phi, \psi)$	$(\text{or } P \ Q)$	
implication	$\phi \Rightarrow \psi$ or $\Rightarrow(\phi, \psi)$	$(\text{implies } P \ Q)$	
equivalence	$\phi \Leftrightarrow \psi$ or $\Leftrightarrow(\phi, \psi)$	$(\text{iff } P \ Q)$	
negation	$\neg\phi$ or $\neg(\phi)$	$(\text{not } P)$	
quantifiers			
universal	$\forall_{x_0 \in X_0, x_1 \in X_1} \phi$ or $\forall_{(x_0 \in X_0, x_1 \in X_1)}(\phi)$	$(\text{forall } ((X0 \ x0) (X1 \ x1)) \ P)$	
existential	$\exists_{x_0 \in X_0, x_1 \in X_1} \phi$ or $\exists_{(x_0 \in X_0, x_1 \in X_1)}(\phi)$	$(\text{exists } ((X0 \ x0) (X1 \ x1)) \ P)$	

Table 4: Syntax Tutorial

namespace, we have used a version of the IFF Top Core (meta) Ontology called the IFF Basic KIF (meta) Ontology (January 2002)⁹. For the diagram and finite-limit namespaces at the type level, we have used the version of the IFF Top Core (meta) Ontology in adjunctive style axiomatization (June 2004)¹⁰. For the meta kernel namespace, and the diagram and finite-limit namespaces at the meta level, we have used a version of the IFF Upper Core (meta) Ontology (January 2002)¹¹. For the exponent namespace at the meta level, we have used the stub for the IFF Upper Core (meta) Ontology in adjunctive style axiomatization (July 2004)¹².

2.2.4. SYNTAX. The LISP Processing (LISP) programming language, which is based on the lambda calculus, is the second oldest (1958) programming language — only the FORTRAN language is older. LISP became the favored programming language for artificial intelligence research. All program code is written as parenthesized lists. The Knowledge Interchange Format (KIF), which has a LISP-like format, was created to serve as a syntax for first-order logic. The IFF logical notation (Table 4), which is a vastly simplified and modified version of KIF, also has a LISP-like format. The IFF grammar¹³ is written in

⁹ located at <http://suo.ieee.org/IFF/metalevel/top/ontology/core/version20020102.pdf>

¹⁰ located at <http://suo.ieee.org/IFF/metastack/TC0.pdf>

¹¹ located at <http://suo.ieee.org/IFF/metalevel/upper/ontology/core/version20020102.pdf>

¹² located at <http://suo.ieee.org/IFF/metastack/UC0.pdf>

¹³ located at <http://suo.ieee.org/IFF/grammar.pdf>

iff:	<pre>(forall ((set ?X) (set ?Y)) (iff (= ?X ?Y) (and (forall ((?X ?x)) (?Y ?x)) (forall ((?Y ?y)) (?X ?y))))))</pre>	#n.set:	<pre>((#n+1).set:set set) ((#n+2).set:subset set (#n+1).set:set) (not (set set))</pre>
type.ftn:	<pre>(iff:set belonging) (forall ((belonging ?xy)) (type.dgm.pr.mor:function-pair ?xy)) (forall ((type.ftn:function ?x) (type.ftn:function ?y)) (iff (belonging [?x ?y]) (exists ((2-cell ?a)) (and (= ?x (source ?a)) (= ?y (target ?a)))))))</pre>	#n.ftn:	<pre>((#n+1).set:set function) ((#n+2).set:subset function (#n+1).ftn:function) ((#n+1).ftn:function source) (= ((#n+1).ftn:source source) function) (= ((#n+1).ftn:target source) #n.set:set) ((#n+2).ftn:restriction source (#n+1).ftn:source) ((#n+1).ftn:function target) (= ((#n+1).ftn:source target) function) (= ((#n+1).ftn:target target) #n.set:set) ((#n+2).ftn:restriction target (#n+1).ftn:target)</pre>
type.pred:	<pre>(iff:set membership) (forall ((membership ?xp)) (exists ((type.ftn:function ?x) (predicate ?p)) (= ?xp [?x ?p]))) (forall ((type.ftn:function ?x) (predicate ?p)) (iff (membership [?x ?p]) (type.ftn:belonging [?x (function ?p)])))</pre>	abc:	<pre>(#n.set:set A) (#n.set:set B) (#n.ftn:function f) (= (#n.ftn:source f) A) (= (#n.ftn:target f) B)</pre>

metashell

natural part

Table 5: Example Code

Extended Backus Naur Form (EBNF). The IFF language contains both logical code and comments. Both nested namespaces and metalevels are specified by prefixes. Table 5 illustrates some IFF code examples. On the left side is code taken from the metashell. The first example taken from the **IFF-IFF** namespace states that two sets are equal precisely when they have the same elements. The second and third examples, taken from the function and predicate subnamespaces of **IFF-TYPE**, define the belonging and membership relations, respectively, as discussed in Section 3 on foundations. On the right side is code taken from the core ontology (**IFF-SET**) in the natural part. This illustrates the chain of code at various metalevels that is necessary in order to declare a level n function $f \in \text{mor}(\text{Set}_n)$. Note that the metashell code is first order, but the natural part code is (almost) atomic, consisting of either declarations, equations or relational expressions (and one negation).

2.2.5. METALANGUAGE. In logic and linguistics, a metalanguage is a language used to make statements about other languages (object languages)¹⁴. Here, we use the term metalanguage to mean a language used to formally express the meaning of another language. An ordered metalanguage is a sequence of metalanguages where each is used as a metalanguage for the next (previous) one. Each level represents a lesser (greater) degree of abstraction. A nested, or hierarchical, metalanguage is an ordered metalanguage where each level is included in (includes) the next one. Each IFF metalevel $1, 2, \dots, n, \dots, \text{meta}, \text{type}, \text{iff}$ services the levels below by providing a metalanguage used to declare and axiomatize those levels. Each of these metalanguages is used to organize

¹⁴Merriam-Webster, Wikipedia

and axiomatize the metalanguages below it. Hence, the IFF is a nested, or hierarchical, metalanguage (see [7] for further discussion)

$$\text{IFF} = \langle \text{IFF}_{\text{iff}} \subset \text{IFF}_{\text{type}} \subset \text{IFF}_{\text{meta}} \subset \cdots \subset \text{IFF}_{n+1} \subset \text{IFF}_n \subset \cdots \text{IFF}_2 \subset \text{IFF}_1 \rangle.$$

The **iff** level metalanguage IFF_{iff} , which is used to talk about (axiomatize) type things, is also used to talk about (axiomatize) meta things and level n things; that is, $\text{IFF}_{\text{iff}} \subset \text{IFF}_{\text{type}}$ ¹⁵. The **type** level metalanguage IFF_{type} , which is used to talk about (axiomatize) meta things, is also used to talk about (axiomatize) level n things; that is, $\text{IFF}_{\text{type}} \subset \text{IFF}_{\text{meta}}$. The level $n+1$ metalanguage IFF_{n+1} , which is used to talk about (axiomatize) level n things, is also used to talk about (axiomatize) level m things for all $m < n$; that is, $\text{IFF}_{n+1} \subset \text{IFF}_n$. In the other direction, level n metalevel namespaces and meta-ontologies use level m terminology and functionality for any metalevel m , where $n < m$; the **meta** namespace uses the **type** and **iff** level terminology and functionality; and the **type** namespace uses the **iff** level terminology and functionality.

3. Foundations

“All of the substance of mathematics can be fully expressed in categories.”

~ F.W. Lawvere

The metashell, core component, and structural component of the natural part provide pointwise (non-traditional) set-theoretic, pointless set-theoretic¹⁶, and category-theoretic foundations for the IFF, respectively. An alternate foundations can be chosen by disconnecting the metashell from the natural part (severing suitable metastack linkage) and discarding it. The core and structural components of the natural part correspond to the distinction between the foundations and organization of mathematics as discussed in the presentation *Kreisel and Lawvere* by J.P. Marquis¹⁷. This component architecture of the IFF, which is forced by the principles of conceptual warrant and categorical design, agrees with Marquis’s claim that “We can have both worlds!”. The foundational approach of the IFF, using core and structural components, possibly addresses some concerns of S. Fefermann [2] concerning the logical and psychological priority of the notions of collection and operation (over categorical notions).

¹⁵The metalanguage IFF_{iff} is very simple, consisting of only the logical symbols in the IFF grammar, plus the five terms `{‘thing’, ‘set’, ‘function’, ‘source’, ‘target’}` defined in the **IFF-IFF** namespace. The metalanguage IFF_{type} is more complicated, consisting of the logical symbols, these five **iff** terms, and around 450 other terms defined in the **IFF-TYPE** namespace that are needed to define a finitely-complete category of abstract sets and functions. The entire IFF metalanguage, which changes as new terminology is added, consists of thousands of terms.

¹⁶Words can have different meanings for different groups in different contexts. The term ‘point’ (or the equivalent term ‘element’) could have a different meaning in the set-theoretic context than it would in the category-theoretic context. The category-theoretic context might regard a morphism to be a (generalized) element, and consider the case of an element of source 1 to be a special element. The set-theoretic context might restrict its notion of element to only these special elements. This is the intent in the jest above.

¹⁷see http://www.math.mcgill.ca/rags/seminar/Marquis_KreiselLawvere.pdf

3.1. MISCONCEPTIONS. Although the IFF is principally oriented towards the roles of applications and support, since the intended support is towards a standard for category theory, the IFF has inevitably found itself involved in the philosophical role of foundations. In this section, our approach to the IFF philosophical role uses the two misconceptions about category theory discussed by F.W. Lawvere in three messages to the CAT list entitled *Why are we concerned?*¹⁸.

Misconception 1: “Category theory is the ‘insubstantial part’ of mathematics and it heralds an era when precise axioms are no longer needed.” This misconception was discussed in Lawvere’s second message.

Misconception 2: “There are ‘size problems’ if one tries to do category theory in a way harmonious with the standard practice of professional set theorists.” This misconception was discussed in Lawvere’s third message.

We handle these in reverse order. The discussion on the Cantorian expansion in subsection 3.2 addresses the second misconception. The discussion on inclusion and membership in subsection 3.3 addresses the first misconception. As a side comment, according to Lawvere in his second message, “Contrary to Fregean rigidity, in mathematics we never use ‘properties’ that are defined on the universe of ‘everything’”. There is the ‘universe of discourse’ principle which is very important: for example, any given group, (or any given topological space, etc.) acts as a universe of discourse.” The IFF syntax addresses this issue. It requires the use of *restricted quantification* in logical expression. For example, the following IFF code axiomatizes the inverse element for a group:

```
(forall ((group ?G))
  (forall ((?G ?a))
    (and (= ((multiplication ?G) [?a ((inverse ?G) ?a)]) (unit ?G))
          (= ((multiplication ?G) [((inverse ?G) ?a) ?a]) (unit ?G)))))
```

3.2. CANTORIAN EXPANSION. The book *Sets for Mathematics* by Lawvere and Rosebrugh [9] has a discussion and a closely reasoned argument (proof) for Cantor’s theorem. In this subsection we extend that argument resulting in a structure we call the Cantorian expansion. The Cantorian expansion provides the metastack, which is the spine of the IFF architecture.

3.2.1. CANTOR’S THEOREM. Let Y be any set. An element $y \in Y$ is a fixed point of an endofunction $\tau : Y \rightarrow Y$ when $\tau(y) = y$. A set Y has the fixed point property when every endofunction on Y has at least one fixed point. Suppose there is a set X and a function $\varphi : X \times X \rightarrow Y$ whose curry $\hat{\varphi} : X \rightarrow Y^X$, where $\hat{\varphi}(a) = \varphi(a, -)$ for all $a \in X$, is surjective; that is, such that for every function $f : X \rightarrow Y$ there is at least one element $a \in X$ such that $f = \hat{\varphi}(a) = \varphi(a, -)$. Then Y has the fixed point property.

Theorem: If a set Y has at least one endofunction $\tau : Y \rightarrow Y$ with no fixed points, then for every set X there is no surjection $X \rightarrow Y^X$. In particular, letting $Y = 2$, since negation $\neg : 2 \rightarrow 2$ has no fixed points, $X < 2^X = \wp X$ for any set X . That is, every set X is strictly smaller than its powerset $2^X = \wp X$.

¹⁸copy located at <http://categorytheorynews.blogspot.com/>

As a corollary, there cannot exist a “universal set” U for which every set X is a subset $X \subseteq U$: if so, then the inclusion $X \rightarrow U$ is an injection; hence, the exponent map $2^U \rightarrow 2^X$ is a surjection; then we can define $X = 2^U$ to get a contradiction. As a further corollary, the collection \mathbf{set} of all sets is not a set: if \mathbf{set} were a set, then $U = \bigcup \mathbf{set}$ would be a “universal set”. There are two essential ingredients for Cantor’s theorem and the above corollaries: existence of the powerset function \wp and the (unbounded) union function \bigcup .

3.2.2. CANTORIAN EXPANSION. The sets here are called “small” sets. The collection of small sets, like the set of natural numbers \mathbb{N} , is either defined naturally, by convention or logically/mathematically¹⁹. The last corollary states that there are sets that are not small. Change the notation, letting \mathbf{set}_1 denote the collection of small sets, and \mathbf{set}_2 denote the collection of sets either small or not (call them “large” sets). So that $\mathbf{set}_1 \subseteq \mathbf{set}_2$, $\mathbf{set}_1 \in \mathbf{set}_2$, but $\mathbf{set}_1 \notin \mathbf{set}_1$. This argument is relative and can be repeated by next using the large sets as the “new” notion of small sets. Hence, starting from the small sets \mathbf{set}_1 , the collection of all sets unfolds into a chain

$$\mathbf{set} = \underbrace{\langle \mathbf{set}_1 \subset \mathbf{set}_2 \subset \cdots \subset \mathbf{set}_n \subset \cdots \rangle}_{\text{Cantorian expansion}}$$

(of Cantorian featureless abstract sets) called the *Cantorian expansion*.

3.2.3. SMALLNESS. The Cantorian expansion is used as the foundation of the IFF. It is the minimal assumption required to obey Cantor’s theorem. In the IFF the collection of small sets is defined by convention²⁰. F.W. Lawvere has suggested at the end of the third email message above that the collection of small sets could be defined logically/mathematically. In particular, that the small sets could be constructed from the large sets as the solution to a certain set isomorphism (equation) expressed using the real numbers. Since the IFF follows only a minimal approach to foundations, it should be possible to further constrain the IFF axiomatization to respect Lawvere’s definition of smallness.

¹⁹The set of natural numbers, which occurs in nature, was used in antiquity (convention) and axiomatized in modern times (logic/math).

²⁰To reiterate, the Cantorian expansion is by convention. First, we assume by convention that a set of “small” sets exists. Call this \mathbf{set}_1 . We also assume there are power and unbounded union operators defined on \mathbf{set}_1 . By Cantor’s theorem $\mathbf{set}_1 \notin \mathbf{set}_1$. Hence, there are sets that are not just small. Call such sets, including \mathbf{set}_1 , “properly large” sets. Call a set, that is either small or properly large, a “large” set. Second, we assume by convention that a set of “large” sets exists. Call this \mathbf{set}_2 . We also assume there are power and unbounded union operators defined on \mathbf{set}_2 . Note that $\mathbf{set}_1 \in \mathbf{set}_2$, $\mathbf{set}_1 \subset \mathbf{set}_2$ and $\mathbf{set}_1 \notin \mathbf{set}_1$. By Cantor’s theorem $\mathbf{set}_2 \notin \mathbf{set}_2$. Hence, there are sets that are not just large. Call such sets, including \mathbf{set}_1 , “properly very large” sets. Call a set, that is either small, large or properly very large, a “very large” set. Third, we assume by convention that a set of “very large” sets exists. Call this \mathbf{set}_3 . We also assume there are power and unbounded union operators defined on \mathbf{set}_3 . Note that $\mathbf{set}_2 \in \mathbf{set}_3$, $\mathbf{set}_2 \subset \mathbf{set}_3$ and $\mathbf{set}_2 \notin \mathbf{set}_2$. By Cantor’s theorem $\mathbf{set}_3 \notin \mathbf{set}_3$. Hence, there are sets that are not just very large. And we continue in this fashion, thereby generating the Cantorian expansion.

3.2.4. UNIONS AND UNIVERSSES. Let n be any metalevel with \mathbf{set}_n the collection of all level n sets²¹. For any level n set $X \in \mathbf{set}_n$, the bounded union operation²² $\bigcup_n^X : \wp\wp X \rightarrow \wp X$ is defined as $\bigcup_n^X(Z) = \{x \in X \mid \exists Y(x \in Y \in Z)\}$ for any family of subsets $Z \in \wp\wp X$. Define $\tilde{\wp}(\mathbf{set}_n) = \mathbf{set}_n \cap \wp\mathbf{set}_n \in \mathbf{set}_{n+1}$ to be the collection of level n sets of level n sets. Then, $\wp\wp X \subseteq \tilde{\wp}(\mathbf{set}_n)$. The unbounded union operation²³ $\bigsqcup_n : \tilde{\wp}(\mathbf{set}_n) \rightarrow \mathbf{set}_n$ is defined as $\bigsqcup_n(Z) = \{x \mid \exists Y(x \in Y \in Z)\}$ ²⁴ for any family $Z \in \tilde{\wp}(\mathbf{set}_n)$. Since $\bigcup_n^X(Z) = \bigsqcup_n Z$ for any $Z \in \wp\wp X$, the bounded union is the restriction of the unbounded union. A level n universe is a level $n+1$ set $U \in \mathbf{set}_{n+1}$ that has the properties: $\mathbf{set}_n \subseteq U$ “every level n set is an element of the universe” and $\mathbf{set}_n \subseteq \wp U$ “every level n set is a subset of the universe”. Then, $\tilde{\wp}(\mathbf{set}_n) \subseteq \wp\wp U$. Define the specific level n universe $\mathbf{univ}_n = \bigsqcup_{n+1} \mathbf{set}_n = \{x \mid \exists Y(x \in Y \in \mathbf{set}_n)\} \in \mathbf{set}_{n+1}$. This defines the special bounded union function $\bigcup_{n+1}^{\mathbf{univ}_n} : \wp\wp\mathbf{univ}_n \rightarrow \wp\mathbf{univ}_n$. Then, the unbounded union of level n sets is the restriction of this bounded union.

3.2.5. GROTHENDIECK UNIVERSSES. The IFF has much in common with Grothendieck universes. A Grothendieck universe \mathcal{U} is meant to provide a set in which all of mathematics can be performed²⁵. The IFF provides a framework in which all of mathematics can be axiomatized. Grothendieck universes model universes of sets. However, IFF universes contain non-set objects such as functions, predicates, relations, vectors, numbers, ships, stars, pelicans and bacteria. This means that Grothendieck universes are more like the toposes $\langle \mathbf{Set}_n, 1 \leq n < \infty \rangle$ than the IFF universes $\langle \mathbf{univ}_n, 1 \leq n < \infty \rangle$. Indeed, the main intuition is that for any set X , there is a Grothendieck universe \mathcal{U} with $X \in \mathcal{U}$. Similarly, for any IFF set X , there is a whole number $1 \leq n < \infty$ with $X \in \mathbf{set}_n = \mathbf{obj}(\mathbf{Set}_n)$. More precisely, a Grothendieck universe \mathcal{U} is a set which is closed under membership and contains doubletons, powers and indexed unions. These axioms imply that a Grothendieck universe \mathcal{U} is closed under the subset order, and contains functions, isomorphs, singletons, indexed coproducts (disjoint unions), indexed products and indexed intersections. Analogs between Grothendieck universes and the IFF are listed in Table 6.

3.3. INCLUSION AND MEMBERSHIP. According to Lawvere, the first misconception “is connected with taking seriously the jest ‘sets without elements’.” We first give the usual definition of basic concepts, such as inclusion and membership, in the category-theoretic context. This follows the usual mathematical practice of defining these relations for subsets of a given universe of discourse represented by an arbitrary category. Then

²¹Following the Cantorian expansion, we assume that \mathbf{set}_n is closed under subset order ($X \in \mathbf{set}_n, Y \subseteq X$ implies $Y \in \mathbf{set}_n$), any level n set is a level $n+1$ set ($\mathbf{set}_n \subseteq \mathbf{set}_{n+1}$), \mathbf{set}_n is itself a level $n+1$ set ($\mathbf{set}_n \in \mathbf{set}_{n+1}$) and \mathbf{set}_n is not a level n set ($\mathbf{set}_n \notin \mathbf{set}_n$).

²²In the current version of the IFF, although we have bounded unions at every metalevel, except for Cantor we have not used either unbounded unions or universes.

²³In the IFF, this union would be specified within and local to a particular IFF metalevel.

²⁴Following Mac Lane in the foundations section of *Categories for the Working Mathematician* [10].

²⁵(by way of Wikipedia) Bourbaki, N., *Univers*, appendix to Expos I of Artin, M., Grothendieck, A., Verdier, J. L., eds., *Théorie des Topos et Cohomologie Étale des Schémas (SGA 4)*, second edition, Springer-Verlag, Heidelberg, 1972.

	Grothendieck universe	IFF
Axioms:	If $X \in \mathcal{U}$ and $x \in X$, then $x \in \mathcal{U}$. If $x, y \in \mathcal{U}$, then $\{x, y\} \in \mathcal{U}$. If $X \in \mathcal{U}$, then $\wp(X) \in \mathcal{U}$. If $I \in \mathcal{U}$ and $X_\alpha \in \mathcal{U}$ for each $\alpha \in I$, then $\bigcup_{\alpha \in I} X_\alpha \in \mathcal{U}$.	If $X \in \mathbf{set}_n$ and $x \in X$, then $x \in \mathbf{univ}_n$. doubleton function $\{-, -\}_X : X^2 \rightarrow \wp(X)$ power set function $\wp : \mathbf{set}_n \rightarrow \mathbf{set}_n$ bounded union $\bigcup_n^X : \wp \wp X \rightarrow \wp X$, and unbounded union $\bigsqcup_n : \tilde{\wp}(\mathbf{set}_n) \rightarrow \mathbf{set}_n$
Theorems:	If $X \in \mathcal{U}$ and $Y \subseteq X$, then $Y \in \mathcal{U}$. If $X, Y \in \mathcal{U}$, then $(f : X \rightarrow Y) \in \mathcal{U}$. If $X \in \mathcal{U}$ and $Y \cong X$, then $Y \in \mathcal{U}$. If $x \in \mathcal{U}$, then $\{x\} \in \mathcal{U}$. If $I \in \mathcal{U}$ and $X_\alpha \in \mathcal{U}$ for each $\alpha \in I$, then $\prod_{\alpha \in I} X_\alpha, \prod_{\alpha \in I} X_\alpha, \bigcap_{\alpha \in I} X_\alpha \in \mathcal{U}$.	If $X \in \mathbf{set}_n$ and $Y \subseteq X$, then $Y \in \mathbf{set}_n$. If $X, Y \in \mathbf{set}_n$, then $(f : X \rightarrow Y) \in \mathbf{ftn}_n$. If $X \in \mathbf{set}_n$ and $Y \cong X$, then $Y \in \mathbf{set}_n$. singleton function $\{-\}_X : X \rightarrow \wp(X)$ The category \mathbf{set}_n is small (co)complete. The preorder $\wp(X)$ is a Boolean algebra.

Table 6: Grothendieck-IFF Analogs

we discuss where these are most generally represented in the IFF.

3.3.1. BASIC DEFINITIONS. Let \mathcal{C} be any category and let X be any \mathcal{C} -object. For any \mathcal{C} -morphism x , x is an element of X , symbolized $x \in X$, when X is the target of x . For any X -element $b \in X$, b is a part of X ²⁶, symbolized $b : X$, when b is a monomorphism. For any two X -elements $x, y \in X$, x belongs to y , symbolized $x \Rightarrow y$, when there exists a proof \mathcal{C} -morphism p such that $x = p \cdot y$ ²⁷. When y is an X -part, the proof p of that belonging is unique. For any two X -parts $a, b : X$, a is included in b , symbolized $a \subseteq b$, when $a \Rightarrow b$. For any X -element $x \in X$ and any X -part $b : X$, x is a member of b , symbolized $x \in b$, when $x \Rightarrow b$. As Lawvere points out, the usual relationship holds between inclusion and membership; that is, inclusion is equivalent to universal implication of membership: $a \subseteq b$ iff $\forall_{x \in X} (x \in a \text{ implies } x \in b)$ for any two X -parts $a, b : X$.

3.3.2. THE TYPE NAMESPACE. The **IFF-TYPE** namespace, a small-sized namespace at the top of the IFF architecture, is at the heart of the IFF metashell. Since the metashell unfolds into the rest of the IFF, we could regard the **IFF-TYPE** namespace, and its kernel in particular, as the heart of the entire IFF. The **IFF-TYPE** namespace is in the middle of the metashell, just below the **IFF-IFF** namespace. But the **IFF-IFF** namespace has little actual content (axiomatization), since it only specifies set and function kind of things. The **IFF-TYPE** namespace is where the axiomatization really begins. It has three nested subnamespaces: the type kernel namespace and namespaces for type diagrams and type limits. The **IFF-TYPE** namespace defines the finitely-complete category $\mathbf{Set}_{\text{type}}$ of abstract sets and functions enriched with factorization and subobjects. In a standard fashion, the finitely-complete category $\mathbf{Set}_{\text{type}}$ defines the bicategory $\mathbf{Span}_{\text{type}} = \mathbf{spn}(\mathbf{Set}_{\text{type}})$ of type sets and type spans, the ordered category $\mathbf{Rel}_{\text{type}} = \mathbf{rel}(\mathbf{Set}_{\text{type}})$ of type sets and type (binary) relations, and their categorical connections. In addition to canonical functionality for finite

²⁶In the IFF, b is called a predicate with genus X .

²⁷Here, composition is written in diagrammatic order.

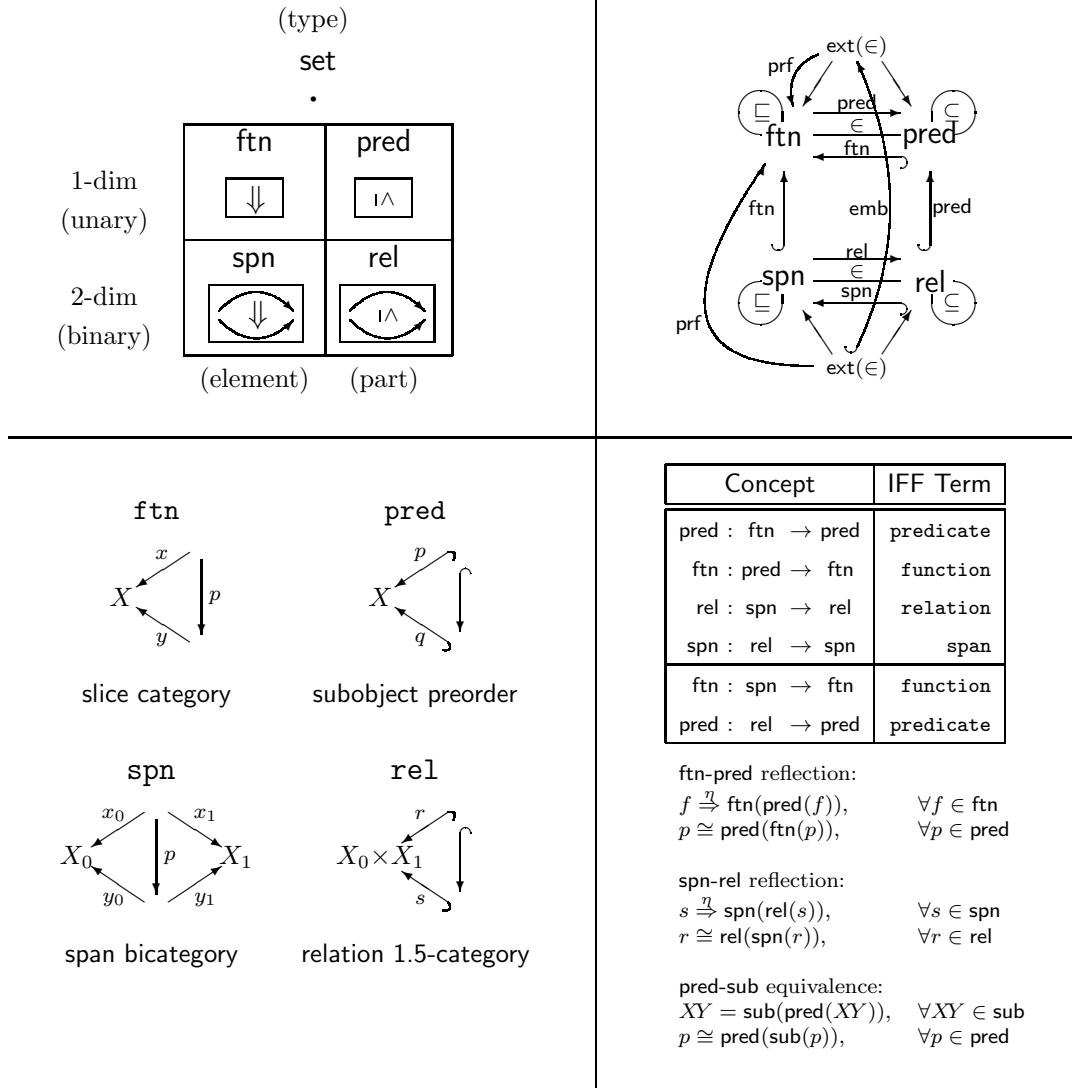


Table 7: IFF Subobject Architecture

limits, the **IFF-TYPE** namespace, specifically its kernel subnamespace, defines (Table 7) four related subobject structures, the belonging relation for functions and spans, and the inclusion relation for predicates and relations. Also defined are membership relations between functions and predicates, and between spans and relations. Associated with these two membership relations are maps that return the proof of membership.

3.3.3. GENERALIZED COMPOSITION. Composition and generalized elements are two essential ingredients in the IFF development. The IFF uses category theory, both in its applications and as a foundation for its architecture. Since composition is fundamental in category theory, it is also fundamental in the use and development of the IFF architecture. In particular, composition for generalized elements is crucially important. Not only do we use the usual composition operation $\circ^n : \mathbf{ftn}_n \times_{\mathbf{set}_n} \mathbf{ftn}_n \rightarrow \mathbf{ftn}_n$ for ordinary point pairs (composable pairs of level n functions) $(f : 1 \rightarrow \mathbf{ftn}_n, g : 1 \rightarrow \mathbf{ftn}_n)$ returning a level n function $f \circ^n g : 1 \rightarrow \mathbf{ftn}_n$, but we also use the parametric composition operation $\hat{\circ}^n$ for generalized point pairs (composable level $n+1$ spans) $(f : \mathbf{ftn}_n \leftarrow X \rightarrow \mathbf{ftn}_n : g)$ for some level n (vertex) set X returning a level $n+1$ function $f \hat{\circ}^n g = (f, g) \circ^{n+1} \circ^n : X \rightarrow \mathbf{ftn}_n$. The use of composition for generalized elements is vital to ensure that the IFF natural part has an atomic expression. Such generalized composition has its most general definition in the meta kernel namespace. This definition uses the type level belonging relation for spans and relations along with its associated proof operation.

4. Future Work

All scientific communities, indeed all communities of discourse (disciplines), create their own conceptual structures with accompanying terminology and meaning (ontologies). Many communities are now working to standardize their ontologies. There is also a search for a unifying framework for these endeavors. It has been suggested [1] that category theory can serve this role — category theory can serve as a meta-ontology, an ontology of ontologies. This was the goal of the SUO IFF project as initially envisioned in the year 2000.

4.1. THE IFF. Following the two guiding principles of conceptual warrant and categorical design, through much experimentation over the last six years the IFF architecture has been developed in essentially a bottom-up fashion. For example²⁸, following the principle of conceptual warrant various ontologies representing first order logic were axiomatized in a first order expression, and then higher level meta-structure was axiomatized which, following the principle of categorical design, would move the axiomatization closer to an atomic expression. During this experimental process²⁹ much logical code was developed for the IFF.

²⁸Actually, the very first IFF ontology developed was the IFF Category Theory meta-ontology **IFF-CAT** [4] located at <http://suo.ieee.org/IFF/metalevel/upper/ontology/category-theory/version20020102.pdf>.

²⁹The motto is that “the IFF is an experiment in foundations”.

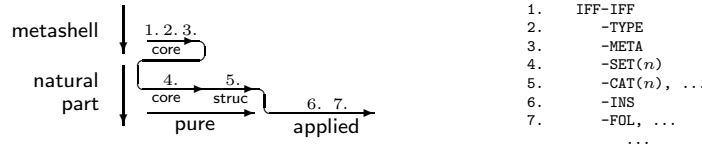


Figure 7: IFF Coding Process Flow

4.1.1. LOGICAL CODING. Recently the final IFF coding was initiated. Great reuse will be made of the experimental code. The coding process flow will follow the architecture in a top-down inside-out fashion as illustrated in Figure 7. The axiomatization (1,2) of the IFF- IFF and IFF- TYPE namespaces is complete. Coding of the IFF- IFF namespace was trivial, since it only involved five terms: ‘thing’, ‘set’, ‘function’, ‘source’ and ‘target’. However because of its importance, the IFF- TYPE namespace that has approximately 450 terms was carefully coded. The next phase of work is (3) the IFF- META namespace. Ignoring exponents, the meta kernel namespace is conceptually the specialization of the type kernel namespace. However, in the IFF- META namespace the axiomatic expressions are flatter and more atomic. Also, the predicate/relation hold expressions, which were defined in the IFF- TYPE namespace, are in great use. The concurrent development processes (illustrated in Figure 4) need to be finished in the meta kernel. After the IFF- META namespace is axiomatized, work will proceed on (4) the generic (level n) set ontology [IFF- SET], and then (5) the structural component of the IFF [IFF- CAT, IFF- 2CAT, IFF- DCAT, etc.]. After the pure aspect is sufficiently coded. work will continue on (6) the ontology axiomatization for institution theory [IFF- INS]. We will also integrate with this (7) previous axiomatizations for logic [IFF- OBJ, IFF- FOL, IFF- IF].

4.1.2. IMPLEMENTATION. Implementation of the IFF will aid the working category theorist. The first step of implementation is finished. This involved writing the IFF grammar. The IFF grammar (written in EBNF)³⁰ works for all of the IFF (both pure and applied aspects, and the metashell). Logical coding of the IFF is based upon this grammar. Also based upon the IFF grammar, we will be able to realize the implementation of a tool suite for the IFF (parsers, viewers, editors, reasoners, etc.). With such an implementation, there could be various application strategies. One strategy might be to represent in the IFF both the syntax and semantics of various other frameworks (common logic, OWL, etc.) in order to absorb any and all of their coded information. But there could be other possible strategies.

4.2. THE COMMUNITY. Much of the previous activities concerning the IFF involved the application of category theory to knowledge engineering, the semantic web, and the representation and semantic integration of ontologies. However, we now want to reverse this approach. We want to apply knowledge engineering to support category theory. A critical element for future work on the IFF involves the development of a standard

³⁰see the IFF syntax document at <http://suo.ieee.org/IFF/grammar.pdf>

ontology for the category theory community.

4.2.1. STANDARDS. A standard is something established by authority, custom, or general consent to be used as a model, example or rule. There are many reasons for standards. Standards allow interoperability between cooperating groups in technology, business and science. Standards are intended to be documented, known descriptions of how something works so that every group who adheres to the standard can interoperate. There is no one true standard. In fact, the evolution of standards is a sign of healthy innovation. The plurality of standards-issuing organizations means that some standards do not necessarily have the support of all communities. The standards of large communities can be created to replace the various incompatible standards of smaller communities.

4.2.2. KINDS. There are several kinds of standards. Two distinctions for standards are open versus proprietary and de facto versus de jure. An open standard is documented for all, and developed and maintained by peers and in the public arena. A proprietary standard is developed and maintained by/for one particular organisation (e.g. Microsoft's Windows, Adobe's PDF). A de facto standard is the property of consortia that represent a wide range of interests. Its status is conferred by use in the marketplace. (e.g. IETF's HTTP protocol, W3C's HTML format, OMG's CORBA) A de jure standard is developed by standards bodies established under national or international laws. It is a well-documented convention, agreed to by participants in a formal standards forum, created through a formal process, and based on the work of a cooperative group or committee of experts. (e.g. the meter of the French Academy of Sciences³¹ and the Geneva Conference on Weights and Measures³², ANSI's C programming language, ISO's JPEG). Also, an ad hoc standard is more widely used than their originator intended (e.g. JVC's VHS, Compuserve's GIF). Various standards project of the Institute of Electrical and Electronics Engineers (IEEE), such as the SUO project, are good examples of the standardization process. The IEEE is heavily involved in developing and maintaining standards for current technologies. Table 8 summarizes the IEEE standization principles and process.

Proposal. *The category theory community will form a working group (under the auspices of some organization or consortium) for the purpose of developing a standard ontology for category theory.*

References

- [1] C.N.G Dampney and Michael Johnson. On category theory as a (meta) ontology for information systems research. In Chris Welty and Barry Smith, editors, *Formal Ontology in Information Systems*, pages 59–69. ACM Press, 2001. Proceedings of

³¹one/ten-millionth of the distance from the equator to the north pole

³²the distance light travels in a vacuum in 1/299,792,458 seconds

Principles <ul style="list-style-type: none"> • <i>due process</i> • <i>openness</i> • <i>consensus</i> • <i>balance</i> • <i>right of appeal</i> 	Process <ol style="list-style-type: none"> 0. idea 1. project approval by standards body 2. draft development by working group <ol style="list-style-type: none"> a. form WG with chair and technical editor b. establish goals, deadlines and schedule c. draft document d. reviewed by technical editor 3. sponsor ballot 4. standards board approval process 5. publish standards 6. periodically reaffirm, revise or withdraw standard; goto 1.
---	--

Table 8: IEEE Standards

the International Conference on Formal Ontology in Information Systems (FOIS'01), Ogunquit, Maine.

- [2] Solomon Feferman. Categorical foundations and foundations of category theory. *Logic, Foundations of Mathematics and Computability Theory*, 1:149–169, 1977.
- [3] Joseph Goguen and Rod Burstall. Institutions: Abstract model theory for specification and programming. *Journal of the Association for Computing Machinery*, 39(1):95–146, January 1992.
- [4] Robert E. Kent. A KIF formalization for the IFF category theory ontology. In *IEEE Standard Upper Ontology workshop at the 17th International Joint Conference on Artificial Intelligence (IJCAI-01), August, 2001, Seattle, Washington*, 2001.
- [5] Robert E. Kent. The IFF foundation for ontological knowledge organization. In Nancy J. Williamson and Clare Beghtol, editors, *Knowledge Organization and Classification in International Information Retrieval*, volume 37 of *Cataloging & Classification Quarterly*, pages 187–203. Haworth, 2003.
- [6] Robert E. Kent. Semantic integration in the IFF. In AnHai Doan, Alon Halevy, and Natasha Noy, editors, *Semantic Integration 2003*, volume 82 of *CEUR Workshop Proceedings*. Sun SITE Central Europe (CEUR), 2003. Proceedings of the Semantic Integration Workshop at ISWC-03, Sanibel Island, Florida, USA, October 20, 2003.
- [7] Robert E. Kent. The Information Flow Framework: A descriptive category metatheory. In *International Category Theory Conference (CT 2004)*, 2004. Slides for the CT 2004 presentation located online: <http://suo.ieee.org/IFF/metastack/CT04%20Presentation.pdf>.
- [8] Robert E. Kent. Semantic integration in the Information Flow Framework. In Y. Kalfoglou, M. Schorlemmer, A. Sheth, S. Staab, and M. Uschold, editors, *Semantic Interoperability and Integration*, number 04391 in Dagstuhl Seminar Proceedings. Dagstuhl Research Online Publication Server, 2005.

- [9] F. William Lawvere and Robert Rosebrugh. *Sets for Mathematics*. Cambridge University Press, 2003.
- [10] Saunders Mac Lane. *Categories for the Working Mathematician*. Springer-Verlag, 1971.
- [11] SUO-WG members. IEEE P1600.1 Standard Upper Ontology Working Group (SUO WG). Technical report, Institute of Electrical and Electronics Engineers, December 2000. Website: <http://suo.ieee.org/>.

Ontologos, 550 SW Staley Dr., Pullman, WA, USA 99163
Email: rekent@ontologos.org